

being suitable for passing scalar values to procedures] ; and

(3) at the client platform, invoking a ["sqlproc"] function [provided by a "client application enablement" (CAE) module, the CAE module] being resident in the client platform [and representing a client component of the DB2 RDBMS] , a parameter list of said [sqlproc] function comprising said [input_args] first parameter and a procedure [_to_invoke] parameter that identifies an user-specified procedure located on the server platform, said [sqlproc] function when executed causing said [input_args] first parameter to be passed to said user-specified procedure at the server platform, and also causing said user-specified procedure to be invoked at the server platform.

2. (Amended) A method of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] the method comprising the steps of:

(1) at the client platform, converting the file to a string;

(2) at the client platform, setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at the server platform;

(3) at the client platform, causing a [pointer in a] first ["sqlvar"] parameter to [point to] represent said string, and causing a [pointer in a] second [sqlvar] parameter to [point to] represent a file name of the file [, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqllda" data type, the sqllda data type being suitable for passing scalar values to procedures] ; and

At Cont.

(4) at the client platform, invoking a ["sqlproc"] function [provided by a "client application enablement" (CAE) module, the CAE module] being resident in the client platform [and representing a client component of the DB2 RDBMS], a parameter list of said [sqlproc] function comprising said [input_args] first and second parameters and said procedure [_to_invoke] parameter, said [sqlproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at the server platform, and also causing said file transfer procedure to be invoked at the server platform.

3. (Amended) The method of claim 2, further comprising the steps of:

- (5) at the server platform, receiving the [input_args] first and second parameters;
- (6) at the server platform, using said [pointer in said] first [sqlvar] parameter [of the input_args parameter] to access and retrieve said string;
- (7) at the server platform, using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name; and
- (8) at the server platform, writing said string to a new file in the server platform using said file name.

4. (Amended) The method of claim 3, further comprising the step of:

- (9) at the client platform, causing a [pointer in a] third [sqlvar] parameter to [point to] represent a target path string.

5. (Amended) The method of claim 4, further comprising the step of:

(10) at the server platform, using said [pointer in said] third [sqlvar] parameter [of the input_args parameter] to access and retrieve said target path string;

wherein step (8) comprises the step of:

at the server platform, writing said string to a new file using said file name at a location in said server platform identified by said target path string.

6. (Amended) A method of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] the method comprising the steps of:

(1) at the server platform, receiving a [n input_args] parameter [being of a "sqlda" data type, said input_args parameter comprising a "sqlvar" parameter defined by said sqlda data type, said sqlvar parameter comprising a pointer pointing to] representing a string that was transferred from the client platform to the server platform, said string being a string representation of a file located on the client platform;

(2) at the server platform, using said [pointer in said sqlvar] parameter [of the input_args parameter] to access and retrieve said string;

(3) at the server platform, writing said string to a new file.

7. (Amended) The method of claim 6, [wherein said input_args parameter] further comprising [es] a second [sqlvar] parameter [having a pointer] that [points to] represents a

file name of said file, and a third [sqlvar] parameter [having a pointer] that [points to]
represents a target path string, the method further comprising the steps of:

(4) at the server platform, using said [pointer in said] second [sqlvar] parameter
[of the input_args parameter] to access and retrieve said file name; and

(5) at the server platform, using said [pointer in said] third [sqlvar] parameter [of
the input_args parameter] to access and retrieve said target path string;

wherein step (3) comprises the step of:

at the server platform, writing said string to a new file using said file name at a location
in said server platform identified by said target path string.

8. (Amended) A method for enabling an user to distribute a procedure from a client
platform to a plurality of server platforms, and for enabling the user to make and register the
procedure at the server platforms, [a "DB2" relational database management system (RDBMS)
executing on each server platform,] comprising the steps of:

(1) at the client platform, displaying a list of servers;
(2) at the client platform, enabling the user to select any of said servers;
(3) at the client platform, commanding a ["client application enablement" (CAE)]
module to distribute a file associated with the procedure to the selected servers, the [CAE]
module being resident in the client platform [and representing a client component of the DB2
RDBMS]; and

(4) at the client platform, commanding the [CAE] module to cause the selected
servers to make and register the procedure;

wherein step (3) comprises the steps of:

- 11
cont.
- (a) converting the file to a string;
- (b) setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at each selected server;
- (c) causing a [pointer in a] first ["sqlvar"] parameter to [point to] represent said string, and causing a [pointer in a] second [sqlvar] parameter to [point to] represent a file name of the file [, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqllda" data type, the sqllda data type being suitable for passing scalar values to procedures] ; and
- (d) invoking, for each selected server, a ["sqleproc"] function provided by the [CAE] module, a parameter list of said [sqleproc] function comprising said [input_args] first and second parameter and said procedure [_to_invoke] parameter, said [sqleproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server.

9. (Amended) The method of claim 8, further comprising the steps of:

- (5) at each server, receiving the [input_args] first and second parameters;
- (6) at each server, using said [pointer in said] first [sqlvar] parameter [of the input_args parameter] to access and retrieve said string;
- (7) at each server, using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name;

(8) at each server, writing said string to a new file in said each server using said file name.

10. (Amended) The method of claim 8, wherein step (4) comprises the step of:
setting the procedure [_to_invoke] parameter equal to information identifying a make function located at each selected server; and

invoking, for each selected server, the ["sqlproc"] function provided by the module, said [sqlproc] function provided by the module when executed causing each selected server to invoke the make function, the make function when executed operating to make the procedure.

11. (Amended) The method of claim 8, wherein step (4) comprises the step of:
setting the procedure [_to_invoke] parameter equal to information identifying a registration function located at each selected server; and

invoking, for each selected server, the ["sqlproc"] function provided by the module, said [sqlproc] function provided by the module when executed causing each selected server to invoke the registration function, the registration function when executed operating to register the procedure [with the DB2 RDBMS] at said each selected server.

12. (Amended) A system of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] the system comprising:

means, at the client platform, for converting the file to a string;

means, at the client platform, for setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at the server platform;

means, at the client platform, for causing a [pointer in a] first ["sqlvar"] parameter to [point to] represent said string, and causing a [pointer in a] second [sqlvar] parameter to [point to] represent a file name of the file, the first and second [sqlvar] parameters [being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures] ; and

means, at the client platform, for invoking a ["sqleproc"] function provided by a ["client application enablement" (CAE)] module [, the CAE module being] resident in the client platform [and representing a client component of the DB2 RDBMS] , a parameter list of said [sqleproc] function comprising [said input_args] first and second parameters and said procedure [_to_invoke] parameter, said [sqleproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at the server platform, and also causing said file transfer procedure to be invoked at the server platform.

13. (Amended) The system of claim 12, further comprising:

means, at the server platform, for receiving the [input_args] first and second parameters;

means, at the server platform, for using said [pointer in said] first [sqlvar] parameter [of the input_args parameter] to access and retrieve said string;

means, at the server platform, for using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name; and;
writing means, at the server platform, for writing said string to a new file in the server platform using said file name.

14. (Amended) The system of claim 13, further comprising:

means, at the client platform, for causing a [pointer in a] third [sqlvar] parameter to point to a target path string.

15. (Amended) The system of claim 14, further comprising:

means, at the server platform, for using said [pointer in said] third [sqlvar] parameter [of the input_args parameter] to access and retrieve said target path string;
wherein said writing means includes means for writing said string to a new file using said file name at a location in said server platform identified by said target path string.

16. (Amended) A server platform coupled to a client platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] the server platform comprising:

means for receiving a [n input_args] first parameter [being of a "sqllda" data type, said input_args parameter comprising a "sqlvar" parameter defined by said sqllda data type, said sqlvar parameter] comprising a [pointer pointing to a] string that was transferred from the client platform to the server platform, said string being a string representation of a file located

on the client platform;

means for using said [pointer in said sqlvar] first parameter [of the input_args parameter] to access and retrieve said string; and

writing means for writing said string to a new file in the server platform.

17. (Amended) The server platform of claim 16, [wherein said input_args parameter] further comprising [es] a second [sqlvar] parameter [having a pointer] that [points to] represents a file name of said file, and a third [sqlvar] parameter [having a pointer] that [points to] represents a target path string, the server platform further comprising:

means for using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name; and

means for using said [pointer in said] third [sqlvar] parameter [of the input_args parameter] to access and retrieve said target path string;

wherein said writing means comprises means for writing said new file using said file name at a location in said server platform identified by said target path string.

SUB B2> 18. (Amended) A [database] system for enabling an user to distribute a procedure from a client platform to a plurality of server platforms, and for enabling the user to make and register the procedure at the server platforms, [a "DB2" relational database management system (RDBMS) executing on each server platform,] the database system comprising:

means, at the client platform, for displaying a list of servers;

means, at the client platform, for enabling the user to select any of said servers;

66040-829330
AA Cont...
[CAE] commanding means, at the client platform, for commanding a ["client application enablement" (CAE)] module to distribute a file associated with the procedure to the selected servers, the [CAE] module being resident in the client platform and representing a client component of the [DB2] RDBMS; and

make and register means, at the client platform, for commanding the [CAE] module to cause the selected servers to make and register the procedure;

wherein said [CAE] commanding means comprises:

means for converting the file to a string;

means for setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at each selected server;

means for causing a [pointer in a] first ["sqlvar"] parameter to [point to] represent said string, and causing a [pointer in a] second [sqlvar] parameter [point to] represent a file name of the file, the first and second [sqlvar] parameters [being part of an input_args parameter, the input_args parameter] having a ["sqllda"] data type [, the sqllda data type] being suitable for passing scalar values to procedures; and

means for invoking, for each selected server, a ["sqlproc"] function provided by the [CAE] module, a parameter list of said [sqlproc] function comprising said [input_args] first and second parameters and said procedure [_to_invoke] parameter, said [sqlproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server.

19. (Amended) The [database] system of claim 18, further comprising:
means, at each server, for receiving the [input_args] first and second parameter;
means, at each server, for using said [pointer in said] first [sqlvar] parameter [of the
input_args parameter] to access and retrieve said string;
means, at each server, for using said [pointer in said] second [sqlvar] parameter [of
the input_args parameter] to access and retrieve said file name; and
means, at each server, for writing said string to a new file in the server using said file
name.

20. (Amended) The system of claim 18, wherein said make and register means
comprises:
means for setting the procedure [_to_invoke] parameter equal to information
identifying a make function located at each selected server; and
means for invoking, for each selected server, the ["sqlproc"] function provided by
the module, said [sqlproc] function provided by the module when executed causing each
selected server to invoke the make function, the make function when executed operating to
make the procedure.

21. (Amended) The system of claim 18, wherein said make and register means
comprises:
means for setting the procedure [_to_invoke] parameter equal to information
identifying a registration function located at each selected server; and

means for invoking, for each selected, server the ["sqlproc"] function provided by the module, said [sqlproc] function provided by the module when executed causing each selected server to invoke the registration function, the registration function when executed operating to register the procedure [with the DB2 RDBMS] at said each selected server.

22. (Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file from a client platform to a server platform, the server platform coupled to the client platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] said method steps comprising:

- (1) at the client platform, converting the file to a string;
- (2) at the client platform, causing a [pointer in an input_args] first parameter to point to said string [, said input_args parameter being of a "sqllda" data type, said pointer being part of a "sqlvar" parameter defined by the sqllda data type, the sqllda data type being suitable by passing scalar values to procedures] ; and
- (3) at the client platform, invoking a ["sqlproc"] function provided by a ["client application enablement" (CAE)] module [, the CAE module being] resident in the client platform [and representing a client component of the DB2 RDBMS] , a parameter list of said [sqlproc] function comprising said [input_args] first parameter and a procedure [_to_invoke] parameter that identifies an user-specified procedure located on the server platform, said [sqlproc] function when executed causing said [input_args] first parameter to be passed to said user-specified procedure at the server platform, and also causing said user-

specified procedure to be invoked at the server platform.

23. (Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file from a client platform to a server platform, the client platform coupled to the server platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] said method steps comprising:

- AI
Cont.
- (1) at the client platform, converting the file to a string;
 - (2) at the client platform, setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at the server platform;
 - (3) at the client platform, causing a [pointer in a] first ["sqlvar"] parameter to point to said string, and causing a [pointer in a] second [sqlvar] parameter to point to a file name of the file[, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqllda" data type, the sqllda data type being suitable for passing scalar values to procedures] ; and
 - (4) at the client platform, invoking a ["sqleproc"] function provided by a ["client application enablement" (CAE)] module [, the CAE module being] resident in the client platform [and representing a client component of the DB2 RDBMS] , a parameter list of said [sqleproc] function comprising said [input_args] first and second parameter and said procedure [_to_invoke] parameter, said [sqleproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at the server platform, and also causing said file transfer procedure to be invoked at the server

platform.

24. (Amended) The program storage device of claim 23, said method steps further comprising the steps of:

- (5) at the server platform, receiving the [input_args] first and second parameters;
- (6) at the server platform, using said [pointer in said] first [sqlvar] parameter [of the input_args parameter] to access and retrieve said string;
- (7) at the server platform, using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name; and
- (8) at the server platform, writing said string to a new file in the server platform using said file name.

25. (Amended) The program storage device of claim 24, said method steps further comprising the step of:

- (9) at the client platform, causing a [pointer in a] third [sqlvar] parameter to point to a target path string.

26. (Amended) The program storage device of claim 25, said method steps further comprising the step of:

- (10) at the server platform, using said [pointer in said] third [sqlvar] parameter [of the input_args parameter] to access and retrieve said target path string;

wherein step (8) comprises the step of:

at the server platform, writing said string to a new file using aid file name at a location in said server platform identified by said target path string.

✓ 27. (Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file from a client platform to a server platform, the client platform coupled to the server platform, [a "DB2" relational database management system (RDBMS) executing on the server platform,] said method steps comprising:

(1) at the server platform, receiving a [n input_args] first parameter [being of a "sqllda" data type, said input_args parameter comprising a "sqlvar" parameter defined by said sqllda data type, said sqlvar parameter] comprising [a pointer pointing to] a string that was transferred from the client platform to the server platform, said string being a string representation of a file located on the client platform;

(2) at the server platform, using said [pointer in said sqlvar] first parameter [of the input_args parameter] to access and retrieve said string;

(3) at the server platform, writing said string to a new file.

28. (Amended) The program storage device of claim 27, [wherein said input_args parameter] further comprising [es] a second [sqlvar] parameter [having a pointer] that [points to] represents a file name of said file, and a third [sqlvar] parameter [having a pointer] that [points to] represents a target path string, said method steps further comprising the steps of:

(4) at the server platform, using said [pointer in said] second [sqlvar] parameter [of the input_args parameter] to access and retrieve said file name; and

(5) at the server platform, using said [pointer in said] third [sqlvar] parameter [of the input_args parameter] to access and retrieve said target path string;

wherein step (3) comprises the step of:

at the server platform, writing said string to a new file using said file name at a location in said server platform identified by said target path string.

29. (Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for enabling an user to distribute a procedure from a client platform to a plurality of server platforms, and for enabling the user to make and register the procedure at the server platforms, [a "DB2" relational database management system (RDBMS) executing on each server platform,] said method steps comprising:

(1) at the client platform, displaying a list of servers;

(2) at the client platform, enabling the user to select any of said servers;

(3) at the client platform, commanding a ["client application enablement" (CAE)] module to distribute a file associated with the procedure to the selected servers, the [CAE] module being resident in the client platform [and representing a client component of the DB2 RDBMS]; and

(4) at the client platform, commanding the [CAE] module to cause the selected servers to make and register the procedure;

wherein step (3) comprises the steps of:

- 11
cont...
- (a) converting the file to a string;
 - (b) setting a procedure [_to_invoke] parameter equal to information identifying a file transfer procedure located at each selected server;
 - (c) causing a [pointer in a] first ["sqlvar"] parameter to point to said string, and causing a [pointer in a] second [sqlvar] parameter to point to a file name of the file [, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures] ; and
 - (d) invoking, for each selected server, a ["sqleproc"] function provided by the [CAE] module, a parameter list of said [sqleproc] function comprising said [input_args] first and second parameters and said procedure [_to_invoke] parameter, said [sqleproc] function when executed causing said [input_args] first and second parameters to be passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server.

30. (Amended) The program storage device of claim 29, said method steps further comprising the steps of:

- (5) at each server, receiving the [input_args] first and second parameters;
- (6) at each server, using said [pointer in said] first [sqlvar] parameter [of the input_args parameter] to access and retrieve said string;
- (7) at each server, using said [pointer in said] second [sqlvar] parameter [of the

input_args parameter] to access and retrieve said file name; and

(8) at each server, writing said string to a new file in said each server using said file name.

31. (Amended) The program storage device of claim 29, wherein step (4) comprises the step of:

setting the procedure [_to_invoke] parameter equal to information identifying a make function located at each selected server; and

invoking, for each selected server, the ["sqleproc"] function provided by the module, said [sqleproc] function provided by the module when executed causing each selected server to invoke the make function, the make function when executed operating to make the procedure.

32. (Amended) The program storage device of claim 29, wherein step (4) comprises the step of:

setting the procedure [_to_invoke] parameter equal to information identifying a registration function located at each selected server; and

invoking, for each selected server, the ["sqleproc"] function provided by the module, said [sqleproc] function provided by the module when executed causing each selected server to invoke the registration function, the registration function when executed operating to register the procedure [with the DB2 RDBMS] at said each selected server.

33. The method of claim 1, wherein said file is a vector file.
34. The method of claim 2, wherein said file is a vector file.
35. The method of claim 6, wherein said file is a vector file.
36. The system of claim 12, wherein said file is a vector file.
37. The program storage device of claim 22, wherein said file is a vector file.
38. The program storage device of claim 23, wherein said file is a vector file.
39. The program storage device of claim 27, wherein said file is a vector file.

SUB B1 40. (New) A method for transferring a file from a client platform to a server platform, comprising:

(a) invoking a function at the client platform that includes one or more parameters to identify a file transfer procedure located at the server platform and to identify the file being transferred; and

(b) invoking the file transfer procedure at the server platform in response to the invoked function at the client platform, wherein the invoked file transfer procedure transfers the file from the client platform to the server platform.

41. (New) The method of claim 40 above, further comprising the step of converting the file being transferred to a string at the client platform and passing the string to the invoked function as a parameter.

42. (New) The method of claim 41 above, further comprising the step of transferring the string to the invoked file transfer procedure at the server platform.

43. (New) The method of claim 42 above, wherein the invoked file transfer procedure at the server platform converts the string to the file.

44. (New) The method of claim 40 above, further comprising the step of transferring the parameters to the server platform.

45. (New) The method of claim 44 above, further comprising the step of transferring the parameters to the invoked file transfer procedure at the server platform.

46. (New) The method of claim 40 above, wherein one or more of the parameters identifies a file name of the file.

47. (New) The method of claim 46 above, wherein the invoked file transfer procedure at the server platform stores the file in the server platform using the file name.

48. (New) The method of claim 40 above, wherein the file comprises a stored procedure.

49. (New) The method of claim 40 above, wherein the file comprises a user defined function.

50. (New) A system for transferring a file from a client platform to a server platform, comprising:

(a) means for invoking a function at the client platform that includes one or more parameters to identify a file transfer procedure located at the server platform and to identify the file being transferred; and

(b) means for invoking the file transfer procedure at the server platform in response to the invoked function at the client platform, wherein the invoked file transfer procedure transfers the file from the client platform to the server platform.

51. (New) The system of claim 50 above, further comprising means for converting the file being transferred to a string at the client platform and for passing the string to the invoked function as a parameter.

52. (New) The system of claim 51 above, further comprising means for transferring the string to the invoked file transfer procedure at the server platform.

53. (New) The system of claim 52 above, wherein the invoked file transfer procedure at the server platform comprises means for converting the string to the file.

54. (New) The system of claim 50 above, further comprising means for transferring the parameters to the server platform.

55. (New) The system of claim 54 above, further comprising means for transferring the parameters to the invoked file transfer procedure at the server platform.

56. (New) The system of claim 50 above, wherein one or more of the parameters identifies a file name of the file.

57. (New) The system of claim 56 above, wherein the invoked file transfer procedure at the server platform stores the file in the server platform using the file name.

58. (New) The system of claim 50 above, wherein the file comprises a stored procedure.

59. (New) The system of claim 50 above, wherein the file comprises a user defined function.

B1
Cont.

60. (New) An article of manufacture comprising a carrier embodying one or more instructions for performing a method for transferring a file from a client platform to a server platform, the method comprising:

(a) invoking a function at the client platform that includes one or more parameters to identify a file transfer procedure located at the server platform and to identify the file being transferred; and

(b) invoking the file transfer procedure at the server platform in response to the invoked function at the client platform, wherein the invoked file transfer procedure transfers the file from the client platform to the server platform.

61. (New) The method of claim 60 above, further comprising the step of converting the file being transferred to a string at the client platform and passing the string to the invoked function as a parameter.

62. (New) The method of claim 61 above, further comprising the step of transferring the string to the invoked file transfer procedure at the server platform.

63. (New) The method of claim 62 above, wherein the invoked file transfer procedure at the server platform converts the string to the file.

64. (New) The method of claim 60 above, further comprising the step of transferring the parameters to the server platform.